AN UNSTEADY SINGLE-PHASE LEVEL SET METHOD FOR VISCOUS FREE SURFACE FLOWS

by

P. M. Carrica, R. V. Wilson, and F. Stern

Sponsored by the Office of Naval Research

Grant N00014-01-1-0073

IIHR Technical Report No. 444

IIHR—Hydroscience & Engineering
The University of Iowa
College of Engineering
Iowa City IA 52242-1585 USA

April 2005

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED
APR 2005		**00-00-2005 to 00-00-2005**

4. TITLE AND SUBTITLE	5a. CONTRACT NUMBER
An Unsteady Single-Phase Level Set Method for Viscous Free Surface Flows	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **The University of Iowa,College of Engineering,IIHR - Hydroscience & Engineering,Iowa City,IA,52242-1585**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES
The original document contains color images.

14. ABSTRACT

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**		**43**	

ABSTRACT

The single-phase level set method for unsteady viscous free surface flows is presented. In contrast to the standard level set method for incompressible flows, the single-phase level set method is concerned with the solution of the flow field in the water (or the denser) phase only. Some of the advantages of such an approach are that the interface remains sharp, the computation is performed within a fluid with uniform properties and that only minor computations are needed in the air. The location of the interface is determined using a signed distance function, and appropriate interpolations at the fluid/fluid interface are used to enforce the jump conditions. A reinitialization procedure has been developed for non-orthogonal grids with large aspect ratios. A convective extension is used to obtain the velocities at previous time-steps for the grid points in air, which allows a good estimation of the total derivatives. In this report we discuss the details of such implementations. The method was applied to three unsteady tests: a plane progressive wave, sloshing in a two-dimensional tank, and the wave diffraction problem in a surface ship, and the results compared against analytical solutions or experimental data. The method can in principle be applied to any problem in which the standard level-set method works, as long as the stress on the second phase can be specified (or neglected) and no bubbles appear in the flow during the computation.

TABLE OF CONTENTS

LIST OF FIGURES

I. INTRODUCTION

Level set methods are becoming increasingly popular for the solution of fluid problems involving moving interfaces [1]. In the case of fluid/fluid interfaces, the level set methods can predict the evolution of complex free surface topologies including waves with larges slopes such as spilling and breaking waves, deforming bubbles and droplets, breakup and coalescence, etc. Since the introduction of the level set method by Osher & Sethian [2], a large amount of bibliography on the subject has been published and several types of problems have been tackled with this method; see for instance the cited review by Sethian & Smereka [1] and the work by Osher & Fedkiw [3].

There is a class of fluid/fluid problems in which the interface between the fluids can be considered as a free-boundary, and therefore the computation can be limited to the more viscous and dense fluid. A most important set of problems of this class is the flow around surface-piercing bodies (like ship hulls) and around submerged bodies (as the flow past a submerged hydrofoil). The idea of solving only the water phase and use of appropriate boundary conditions at the free surface is not new, and has been used with almost any interface tracking method. Volume of Fluid (VOF) methods solving only the water (or the denser) phase are common (see for instance [4]).

Solving only the water phase in level set methods (here called the single-phase level set method) presents several advantages over the classic level set approach in which both fluids are solved (or two-phase level set method). One of them is that in the air phase only extension velocities are needed, which makes the problem considerably easier to converge. In addition, since we are solving on a single fluid with constant properties, the computation of the pressure can be done in a standard way without pressure and velocity oscillations at the interface that are common in two-phase level set methods with large density ratios.

We stress that in principle the single-phase level-set method can handle any problem that can be solved with two-phase level-set methods, as long as the two conditions discussed next are met. The first condition arises from the fact that in single-phase level set methods (or any other method in which we solve only the water phase) the continuity condition will not be satisfied on the air phase, thus the method is not suitable for problems in which the air phase somehow gets pressurized during the computation.

This means that the method will yield non-physical results if air is trapped or bubbles are formed inside the liquid during the calculation. The second condition, related to the first, is that the stresses caused on the liquid phase by the air phase must be negligible since, again, we don't compute in the air and impose those stresses to be zero, or specified to some value if we model, for instance, breaking waves or wind-induced stresses. Other than these limitations, the method has no restrictions on surface topology, allowing for large amplitude and/or very steep waves.

The main application we pursue in our research program is the computation of the viscous free surface flow in large surface-piercing bodies. This is a difficult problem that involves complicated three-dimensional geometries, which leads to non-orthogonal curvilinear grids with high aspect ratio, very high Reynolds numbers and complex free surface topologies. Both surface tracking and surface capturing methods have been used to tackle this type of problems.

In surface tracking methods, the computational grid is fitted to the free surface, and therefore is not fixed in time. This type of method can be high order accurate both in space and time. The grid deformation process to fit the grid to the free surface works well and is robust as long as the free surface slope remains small. Examples of successful applications for free surface ship flows include resistance computation without [5] and with propellers [6], forward speed diffraction [7,8], roll decay motion [9] and pitching and heaving motion in regular head seas [10]. Unfortunately, as the deformation of the free surface increases it is difficult to prevent grid quality deterioration and computation breakdown. One of the main conclusions drawn after the Gothenburg 2000 workshop on ship hydrodynamics [11] was that level-set methods show promising results and further research for surface capturing methods should be pursued to overcome the limitations of surface-tracking methods.

Besides level set methods, surface capturing methods include volume of fluid (VOF) methods [12], front tracking methods [13], and other variations in which a color or volume fraction function is computed. Among others, notable examples of the application of surface capturing methods for flows around floating bodies are in [14], where a surface capturing method is used to simulate the flow around a surface piercing blunt body, and in Sato *et al*. [15], who studied pitching and heaving on linear incident waves. Another

surface capturing method for ships, based on a gas volume fraction approach, is presented in [16].

Level set methods have been applied for both submerged and surface-piercing body problems. [17] used both single and two-phase level set methods to study the flow around submerged two-dimensional hydrofoil in steady-state. In [18,19,20], the authors solve more complicated three-dimensional flows around container ships using the two-phase level set approach with body-fitted coordinates. Three-dimensional flows around ships have been solved in steady-state using a single-phase level set method [21]. We note that all these single-phase methods are devised and used for steady-state applications only.

In this paper we present an unsteady single-phase level set method for viscous, incompressible flow. The computation of the total time derivative is a key issue to make the method time-accurate. The implementation of the overall scheme is discussed.

We test the method against two two-dimensional and one three-dimensional unsteady cases: a linear progressive wave, the sloshing in a steady tank, and the wave diffraction by a ship. Results are compared against analytical or experimental data showing good agreement.

II. SINGLE- AND TWO-PHASE LEVEL SET METHODS

The standard level set method for incompressible free surface viscous flows originated about ten years ago [22] and has become very popular. We call this method the two-phase level set, since the solution is obtained in both fluids as discussed below.

In a two-phase flow, the instantaneous local equations of motion within each fluid can be written as [23]:

$$\frac{\partial \mathbf{v}_k}{\partial t} + \mathbf{v}_k \cdot \nabla \mathbf{v}_k = -\frac{1}{\rho_k} \nabla p_k + 2\frac{\mu_k}{\rho_k} \nabla \cdot \mathbf{D}_k + \mathbf{g} \qquad (1)$$

$$\nabla \cdot \mathbf{v}_k = 0 \qquad (2)$$

where the subscript $k = l$ or g indicates the phase present at a given point in space, which in our case can be either liquid or gas. $\mathbf{v}$, p and $\mathbf{D}$ are the velocity, pressure and rate of deformation tensor, ρ_k and μ_k are the density and viscosity of fluid k, and $\mathbf{g}$ is the

gravity acceleration. At the interface, the interfacial boundary conditions or jump conditions apply. In the case of immiscible fluids we can write:

$$\left[-p\,\mathbf{I}+2\,\mu\,\mathbf{D}\right]\cdot\mathbf{n}=-\left(\sigma\,\kappa\,\mathbf{n}+\nabla_i\sigma\right) \tag{3}$$

where the bracket means $l - g$ and the normal $\mathbf{n}$ is taken from the liquid to the gas. The second term on the right hand side of Eq. (3) is the stress due to gradients on surface tension or Marangoni effect [24], usually important when large gradients of temperature are present, as in boiling flow. ∇_i denotes the gradient in the local free surface coordinates. The interfacial curvature κ is computed from:

$$\kappa=\nabla\cdot\mathbf{n} \tag{4}$$

The jump conditions (3) can be integrated into the equations of motion (1) and (2), resulting in a body force concentrated in the interface of a single incompressible fluid with variable properties [25]:

$$\frac{\partial\,\mathbf{v}}{\partial t}+\mathbf{v}\cdot\nabla\mathbf{v}=-\frac{1}{\rho}\nabla p+2\frac{\mu}{\rho}\nabla\cdot\mathbf{D}+\mathbf{g}+\left(\sigma\,\kappa\,\mathbf{n}+\nabla_i\sigma\right)\delta\left(\phi\right) \tag{5}$$

where ϕ is a distance to the interface function, positive in liquid and negative in gas. The location of the interface is then given by the zero level set of the function ϕ, known as the level set function. Since the free surface is a material interface (in absence of interfacial mass transfer such as evaporation or condensation), then the equation for the level set function is:

$$\frac{\partial\,\phi}{\partial\,t}+\mathbf{v}\cdot\nabla\phi=0 \tag{6}$$

and from the level set function we can compute the normal as:

$$\mathbf{n}=-\frac{\nabla\phi}{\left|\nabla\phi\right|} \tag{7}$$

Since the fluid properties in Eq. (5) change discontinuously across the interface, and the concentrated surface tension force also becomes infinite in an infinitesimal volume, direct solution of Eq. (5) is naturally difficult. Two approaches are usually followed to overcome these difficulties.

In the standard level set method, the interface is smoothed across a finite thickness region, usually a few grid points thick. The fluid properties and the delta function are thus modified as [21]:

$$\rho(\phi) = \rho_g + (\rho_l - \rho_g) H(\phi) \tag{7a}$$

$$\mu(\phi) = \mu_g + (\mu_l - \mu_g) H(\phi) \tag{7b}$$

$$\delta(\phi) = \frac{d H(\phi)}{d \phi} \tag{7c}$$

where the smoothed Heaviside function is usually expressed as:

$$H(\phi) = \begin{cases} 0 & (\phi < -\alpha) \\ 0.5\left[1 + \phi/\alpha + \sin(\pi\phi/\alpha)/\pi\right] & (|\phi| \le \alpha) \\ 1 & (\phi > \alpha) \end{cases} \tag{8}$$

with α the half-thickness of the properties transition region. An important step is to maintain the level set function a distance function within the transition region at all times. This is achieved by the reinitialization step, discussed later in this paper.

One drawback of the level set method is the introduction of the transition region. This results in smearing of the flow properties and variables, forcing them to be continuous at the interface regardless of the appropriate jump conditions. This problem is solved on the Ghost Fluid Method, in which the jump conditions are introduced implicitly on the formulation by solving for a "ghost" fluid across the interface [26,27]. This approach, though easy to implement, forces the solution of two fluid fields on each grid node, one for each fluid. This results in additional computational cost that can be very demanding in large 3-D computations.

It is then desirable for many applications to be able to solve a single-phase problem in a fixed grid, capturing the interface with appropriate enforcement of the jump conditions, and still retaining the advantages of level set methods. These applications generally involve air-water flows in which the density and viscosity ratios are about 1000 and 75, respectively. Under these conditions, the interface can be taken as shear stress free for most applications, as frequently done with interface tracking algorithms. In this way the computational domain to solve the RANS equations is restricted to the grid points in water plus a few nodes in air to enforce the jump conditions, with the

consequent economy of resources. A second advantage is that the continuity equation is enforced always in a single fluid, thus allowing the use of standard collocated methods without the usual pressure and velocity oscillations that occur at the interface between fluids with a large density ratio ([1], p. 355). Such a method is the single-phase level set method [17,21]. On the next sections we present an extension to the single-phase level set method in order to make it appropriate for unsteady problems.

III. UNSTEADY SINGLE-PHASE LEVEL SET METHOD DETAILS

In this section we describe the details of the single-phase level-set method, including derivation and implementation of the jump conditions, reinitialization of the level-set as a distance function and computation of the total time derivatives.

A. Governing Equations

Since we will solve the problem on a single fluid, the RANS equations can be non-dimensionalized as usual to obtain:

$$\frac{\partial \mathbf{v}}{\partial t} + \mathbf{v}\cdot\nabla\mathbf{v} = -\nabla p + \nabla\cdot\left(\frac{1}{\mathrm{Re}_{eff}}\nabla\mathbf{v}\right) + \mathbf{S} \tag{9}$$

where the source **S** includes any volumetric source with the exception of the gravity, which is lumped with the pressure to define the non-dimensional piezometric pressure:

$$p = \frac{p_{abs}}{\rho\ U_0^2} + \frac{z}{Fr^2} \tag{10}$$

In Eqs. (9) and (10) p_{abs} is the absolute pressure, Re_{eff} is the effective Reynolds number and *Fr* is the Froude number, defined as:

$$\mathrm{Re}_{eff} = \frac{U_0 L}{\nu + \nu_t} \tag{11}$$

$$Fr = \frac{U_0}{\sqrt{g L}} \tag{12}$$

where U_0 is the free-stream velocity and *L* is the characteristic length. ν_t is the turbulent viscosity, that in our approach, is obtained after solving the blended $k-\omega$ model of

turbulence [28]. Since we are considering an incompressible fluid, the continuity equation reads:

$$\nabla \cdot \mathbf{v} = 0 \tag{13}$$

To capture the location of the interface, we solve the level set function, Eq. (6).

B. Enforcement of the Jump Conditions

In contrast to the two-phase level-set method, in which the interfacial jump conditions are embedded naturally on the formulation, the jump conditions at the interface between two fluids, Eq. (3), must be treated as a boundary condition enforced explicitly in a single-phase level set approach. The jump condition on any direction tangential to the free surface is [27]:

$$\left[\mu \left(\nabla \mathbf{v} \cdot \mathbf{n} \right) \cdot \mathbf{t} + \mu \left(\nabla \mathbf{v} \cdot \mathbf{t} \right) \cdot \mathbf{n} \right] = 0 \tag{14}$$

Neglecting the viscosity in air, and since **t** is an arbitrary vector perpendicular to the normal to the interface, we obtain the boundary conditions for the velocity at the interface:

$$\nabla \mathbf{v} \cdot \mathbf{n} = 0 \tag{15}$$

In the direction normal to the interface, the jump condition in dimensional form can be written as:

$$\left[p_{abs} - 2\mu \left(\nabla \mathbf{v} \cdot \mathbf{n} \right) \cdot \mathbf{n} \right] = \sigma \kappa + \nabla_i \sigma \cdot \mathbf{n} \tag{16}$$

As a good approximation in water/air interfaces, the pressure can be taken as constant in the air. Also, because of Eq. (15), the second term on the left hand side of Eq. (16) is zero. Thus the jump condition reduces to:

$$p_{abs} = \sigma \kappa + \nabla_i \sigma \cdot \mathbf{n} \tag{17}$$

We also choose at this point to neglect the surface tension effects, because for the class of problems in which we are interested the curvature of the free surface is small. This is not a limitation of the model and surface tension can be easily included, as done by Di Mascio *et al*. [21]. Introducing the dimensionless piezometric pressure, Eq. (10), we must impose at the fluid/fluid interface:

$$p = \frac{z}{Fr^2} \tag{18}$$

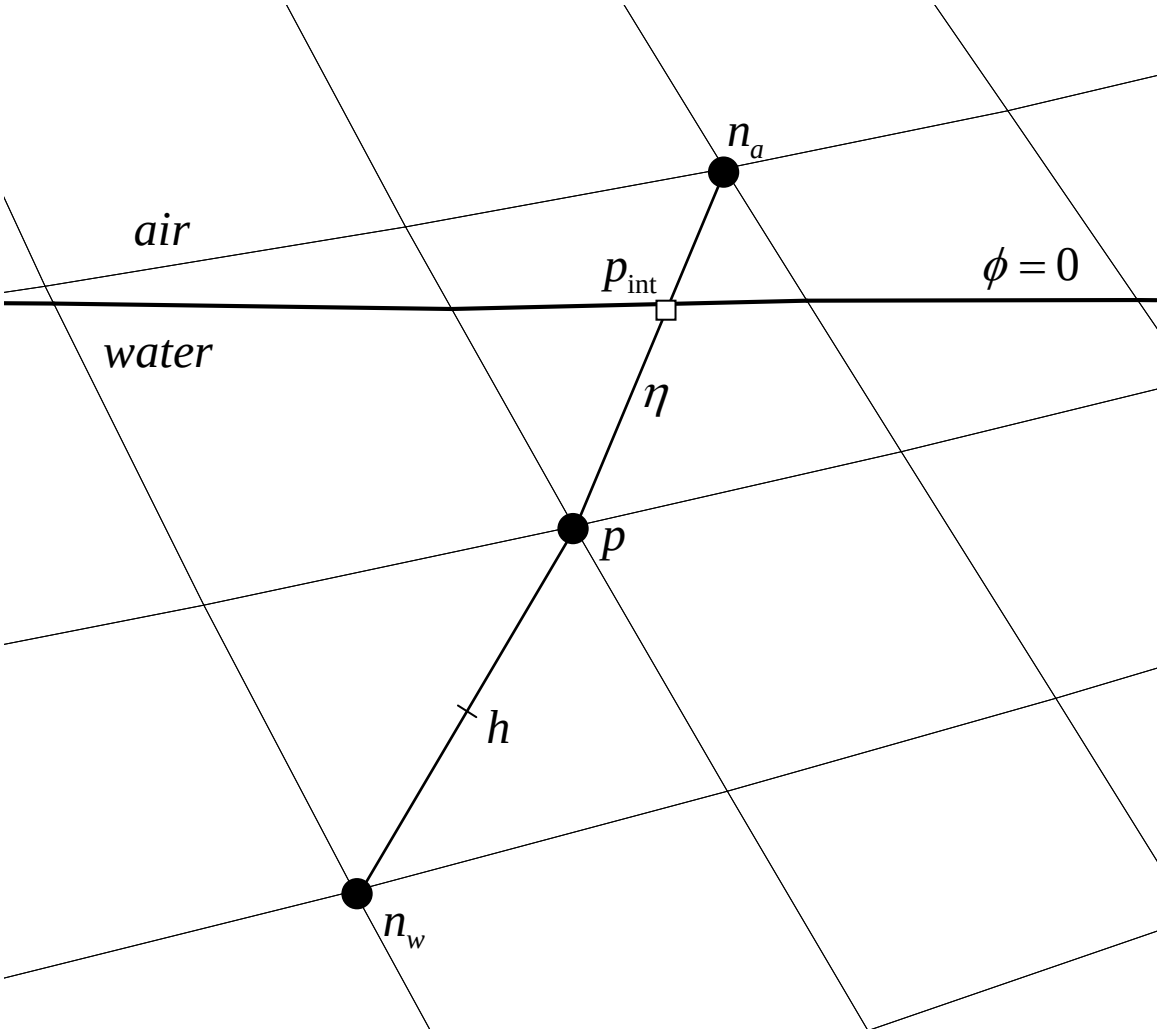

Figure 1: Computation of the free surface location and calculation of the neighbor pressure to enforce the pressure boundary condition.

C. Pressure Condition at the Free Surface

The pressure at the free surface is given by Eq. (18) and was derived from the jump conditions. In a surface capturing approach, however, the free surface is not located at the grid points and therefore an interpolation scheme must be devised to enforce the interfacial pressure condition at the interface location. The free surface itself can be easily identified by locating the change in sign of ϕ between two contiguous grid points along any coordinate line.

We have then in our method three types of grid points. Grid points in water with all the first neighbors in water will be computed without any special treatment and need

no additional consideration. Grid points in air can have any pressure value, and then we choose to enforce there Eq. (18) where z is now the vertical coordinate at the point. For points in water in which at least one of the neighbors is in air the following discussion applies.

For any grid point p in water that has a neighbor in air n_a, the interfacial pressure condition of Eq. (18) is enforced locally. Referring to Fig. 1, the relative distance between the grid point in water and the interface is:

$$\eta = \frac{\phi_p}{\phi_p - \phi_{na}} \tag{19}$$

thus, interpolating along the line joining the points p and n_a and using Eq. (18) we obtain for the interfacial pressure:

$$p_{\text{int}} = \frac{(1-\eta) z_p + \eta z_{na}}{Fr^2} \tag{20}$$

The pressure at the neighbor in air can then be found by extrapolation from the pressure values at the points h and at the interface, where h is located halfway between the local point p and the opposite neighbor in water to n_a, shown as n_w in Fig. 1:

$$p_{na} = (p_{\text{int}} - p_h) \frac{dist(\mathbf{r}_{na}, \text{int})}{dist(\mathbf{r}_p, \text{int}) + dist(\mathbf{r}_p, \mathbf{r}_h)} + p_{\text{int}} \tag{21}$$

where $p_h = (p_p + p_{nw})/2$ and $\mathbf{r}_h = (\mathbf{r}_p + \mathbf{r}_{nw})/2$. This scheme can be easily implemented on a subroutine that computes the matrix coefficients for the pressure Poisson equation, by plugging Eq. (21) on every neighbor in air. Since all the necessary neighbors to enforce the interfacial pressure condition are the same necessary to build the pressure matrix (for typical 19 point stencils), no additional connectivity has been added and therefore can be implemented in multi-block codes with no modifications on the existing inter-block information transfer scheme. This is especially attractive in parallel implementations.

D. Reinitialization

In two-phase level set methods, the distance function is reinitialized periodically to keep it a smooth distance function and have a transition region uniform in thickness. The reinitialization step is extremely important in single-phase level set methods but for

different reasons: the normal must be accurately evaluated at the interface because it is used in the boundary conditions and it must also be reasonable everywhere in air since it is used to extend the velocities into the air to transport the level set function.

In our case we split the reinitialization procedure in two steps. The first step is a close point reinitialization for those grid points that are neighbors to the interface as in the Fast Marching Method [29]. In the second step, we solve a transport equation for the rest of the grid points, using the near-boundary points as Dirichlet boundary conditions. This is easy to implement in parallel environments and reasonably inexpensive.

We extend the method of Adalsteinsson & Sethian [30] to three dimensional curvilinear grids to obtain a good signed distance for the first neighbors to an interface (the beginning set of their *close* points). In curvilinear grids with very large aspect ratio (in boundary layer grids can be as large as 10^5) we cannot use the distance to the first neighbor to define the geometrical distance to the interface. In Fig. 2 we show such a grid, marking with circles all points to be geometrically reinitialized because they have at least a first neighbor in a different fluid. Consider the point that is filled with white. For that point the closest interface lies on the $+\eta$ direction before crossing the third neighbor. If we were to reinitialize using only first neighbors we would only find the interface on the $-\xi$ direction and a poor signed distance would result. This problem becomes unacceptable for very large aspect ratios, where the closest interface might lie several grid points away in some direction. Moreover, the closest interface might be in a different block, and thus that information belongs to a different processor in a typical domain decomposition parallel implementation.

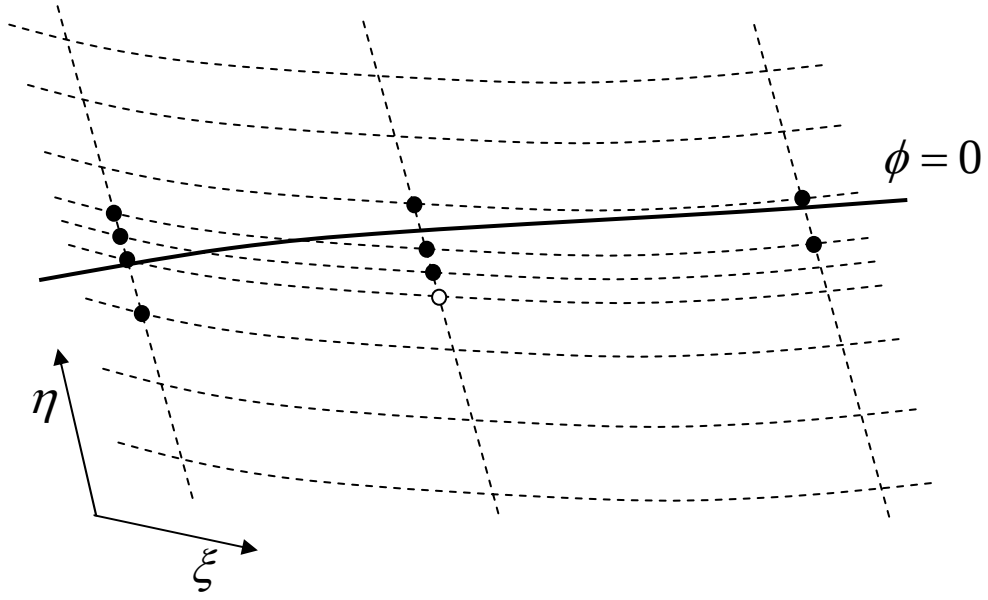

Figure 2: Reinitialization of the close points.

Our close point reinitialization algorithm searches along the grid lines, including neighbor blocks, in each direction, and finds the location of the intersection of the grid lines with the interface. Let these locations be $\mathbf{r}_{\xi+}, \mathbf{r}_{\xi+}, \mathbf{r}_{\eta+}, \mathbf{r}_{\eta-}, \mathbf{r}_{\zeta+}, \mathbf{r}_{\zeta-}$, where we must note that in some directions there might be no interfaces. The distance from the point to be reinitialized to the interface is given by the distance to the plane formed by the three points:

$$\mathbf{r}_\xi = \min\left(\mathbf{r}_{\xi+}, \mathbf{r}_{\xi-}\right), \quad \mathbf{r}_\eta = \min\left(\mathbf{r}_{\eta+}, \mathbf{r}_{\eta-}\right), \quad \mathbf{r}_\zeta = \min\left(\mathbf{r}_{\zeta+}, \mathbf{r}_{\zeta-}\right) \tag{22}$$

$$d = \frac{\left[\left(\mathbf{r}_\xi - \mathbf{r}_p\right) \times \left(\mathbf{r}_\eta - \mathbf{r}_p\right)\right] \cdot \left(\mathbf{r}_\zeta - \mathbf{r}_p\right)}{\left|\left(\mathbf{r}_\xi - \mathbf{r}_\eta\right) \times \left(\mathbf{r}_\xi - \mathbf{r}_\zeta\right)\right|} \tag{23}$$

If along any of the gridlines no interface is found then we compute the distance to a line if we found two interfaces or to a point if we found only one.

This algorithm, though not prohibitive, is not very fast because necessarily involves a much careful search than that presented in [30]. Thus for the second step of the reinitialization and for the extension, Eq. (15), we chose to solve a PDE. For the reinitialization we have:

$$\mathbf{n} \cdot \nabla \phi = sign\left(\phi_0\right) \tag{24}$$

where **n** is in this case the normal pointing to the fluid being reinitialized. Since **n** is either $\nabla\phi$ or $-\nabla\phi$, Eq. (24) is an eikonal equation and propagates information from the interface outwards. The corresponding Dirichlet boundary conditions for (24) are given by the value of ϕ at the close points. Notice also that as written, Eq. (24) is nonlinear and requires some iterations to converge, though its solution is very fast in the overall scheme.

E. Computing the Total Time Derivatives

The proper computation of the total time derivative terms near the interface is a problem in single-phase level set techniques. To understand this, consider an interface moving vertically, as shown in Fig. 3, where we want to compute the total derivative at point *p*. At the current time shown, the point *p* is in water and in the previous time step was in air. Since we are using an Eulerian approach, the local time derivative at point *p* is expressed as a first-order approximation as:

$$\frac{\partial\varphi}{\partial t} \cong \frac{\varphi_t - \varphi_{t-\Delta t}}{\Delta t} \tag{25}$$

where φ is any velocity or turbulence variable. The problem appears because $\varphi_{t-\Delta t}$ was in air on the previous time step and therefore its value does not satisfy the field equations since it was computed using the extension of Eq. (15). In [17,21] the authors recognize this limitation and deal with steady-state problems only. Notice that in the standard two-phase level set method we can have a problem of similar origin.

Assume for instance that our level set method allows completely sharp interfaces and that in Fig. 3 the air is moving from left to right and the liquid is moving vertically, which is allowed because the stress and velocities must be continuous at the interface but not between contiguous grid points crossing the interface. Then when liquid reaches a point that before was in air it will have the information that in the previous time steps it was moving from left to right. This results in a spurious inertia and consequently the liquid velocity will gain some left to right component. This effect is highly limited in level set methods by the thickness of the transition region and the limitations on the time step in explicit methods that prevent large changes on the interface location between time steps.

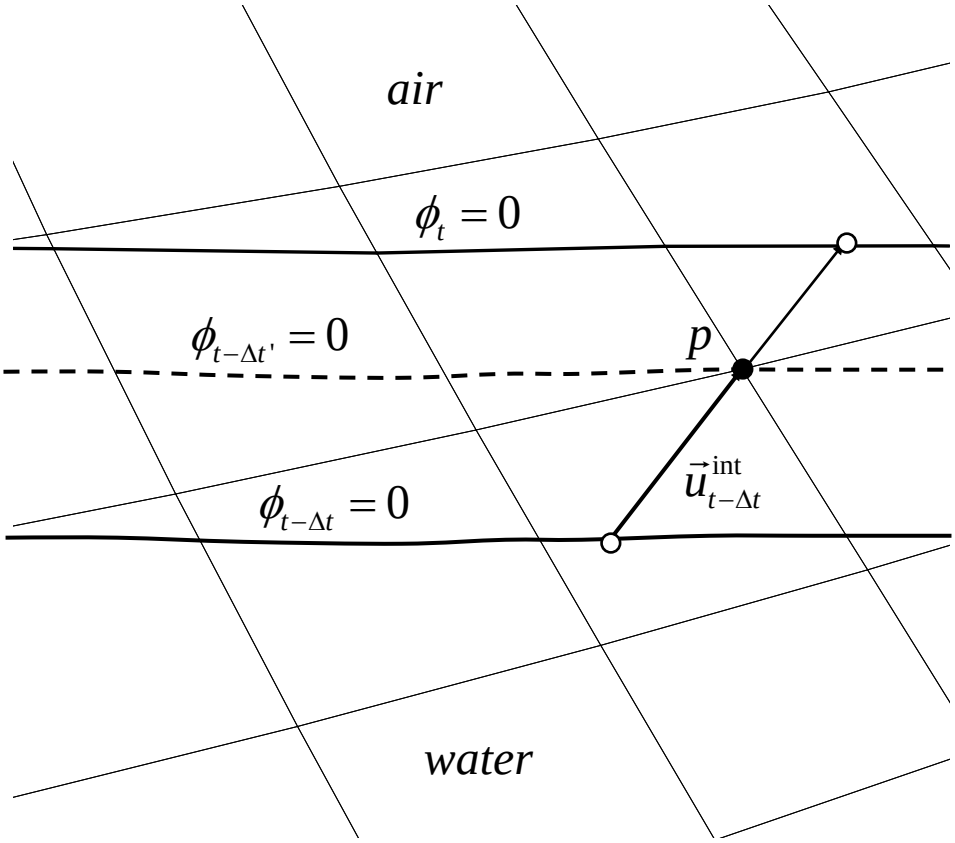

Figure 3: Computation of the time derivative at a grid point that is changing fluid.

We use for our analysis the velocities, though all the variables should be computed across the interface on the same way.

In single-phase level set approaches we have a sharp interface, and thus we have to find a suitable velocity for the previous time-steps to assign to the grid point previously in air. One possibility is to use the extended velocity as the previous time-step velocity (or time steps in a higher order scheme). However, this extended velocity will not result in the right total derivative. An extension that will yield a good approximation of the total derivative is presented below.

Let's consider a particle belonging to the free surface that at time $t-\Delta t'$ during the time step advancing from $t-\Delta t$ to t crosses a grid point, marked as *p* in Fig. 3. From $t-\Delta t'$ to *t*, the total derivative of the velocity is:

$$\left.\frac{D\mathbf{u}}{Dt}\right|_1 = \frac{\mathbf{u}\left(\mathbf{r}_p, t-\Delta t'\right) - \mathbf{u}_{t-\Delta t}^{\text{int}}}{\Delta t - \Delta t'} \tag{26}$$

where $\mathbf{u}\left(\mathbf{r}_p, t-\Delta t'\right)$ is the velocity of the particle on the interface that at $t-\Delta t'$ is exactly on grid point *p*, and $\mathbf{u}_{t-\Delta t}^{\text{int}}$ is the velocity of the same particle at $t-\Delta t$. This is a Lagrangian evaluation of the acceleration.

From $t-\Delta t'$ to *t* we can use the Eulerian acceleration, since from that time on the grid point *p* is in water:

$$\left.\frac{D\mathbf{u}}{Dt}\right|_2 = \frac{\mathbf{u}\left(\mathbf{r}_p, t\right) - \mathbf{u}\left(\mathbf{r}_p, t-\Delta t'\right)}{\Delta t'} + \mathbf{u}\left(\mathbf{r}_p, t\right)\cdot\nabla\mathbf{u}\left(\mathbf{r}_p, t\right) \tag{27}$$

Since both total derivatives are computed within the same time step advancing from $t-\Delta t$ to *t*, we set $\left.\frac{D\mathbf{u}}{Dt}\right|_1 = \left.\frac{D\mathbf{u}}{Dt}\right|_2$ and solve for the unknown velocity $\mathbf{u}\left(\mathbf{r}_p, t-\Delta t'\right)$ to obtain:

$$\frac{D\mathbf{u}}{Dt} = \frac{\mathbf{u}\left(\mathbf{r}_p, t\right) - \mathbf{u}_{t-\Delta t}^{\text{int}}}{\Delta t} + \frac{\Delta t'}{\Delta t}\mathbf{u}\left(\mathbf{r}_p, t\right)\cdot\nabla\mathbf{u}\left(\mathbf{r}_p, t\right) \tag{28}$$

Notice that the ratio $\Delta t'/\Delta t$ can be computed from the level set function as:

$$\frac{\Delta t'}{\Delta t} = \frac{\phi\left(\mathbf{r}_p, t\right)}{\phi\left(\mathbf{r}_p, t\right) - \phi\left(\mathbf{r}_p, t-\Delta t\right)} \tag{29}$$

Thus, accordingly to Eqs. (28) and (29), the total time derivative in grid points in which the level set function changes from air to water is replaced by:

$$\frac{D\mathbf{u}}{Dt} = \frac{\mathbf{u}\left(\mathbf{r}_p, t\right) - \mathbf{u}_{t-\Delta t}^{\text{int}}}{\Delta t} + \frac{\phi\left(\mathbf{r}_p, t\right)}{\phi\left(\mathbf{r}_p, t\right) - \phi\left(\mathbf{r}_p, t-\Delta t\right)}\mathbf{u}\left(\mathbf{r}_p, t\right)\cdot\nabla\mathbf{u}\left(\mathbf{r}_p, t\right) \tag{30}$$

where the interfacial particle velocity at $t-\Delta t$, $\mathbf{u}_{t-\Delta t}^{\text{int}}$, remains to be determined. Notice that if in Eq. (30) we replace this velocity by the velocity at point *p* on the previous time step we obtain the following expression:

$$\frac{D\mathbf{u}}{Dt} = \frac{\mathbf{u}\left(\mathbf{r}_p, t\right) - \mathbf{u}\left(\mathbf{r}_p, t-\Delta t\right)}{\Delta t} + \frac{\phi\left(\mathbf{r}_p, t\right)}{\phi\left(\mathbf{r}_p, t\right) - \phi\left(\mathbf{r}_p, t-\Delta t\right)}\mathbf{u}\left(\mathbf{r}_p, t\right)\cdot\nabla\mathbf{u}\left(\mathbf{r}_p, t\right) \tag{31}$$

The implementation of Eq. (31) requires an easy modification to the convective term and that we load on the points in air the interfacial velocity of the particle that will pass through grid point *p*, and assign that to the previous time step velocity. This second step is achieved by simply extending the variables not with the normal but with the velocity itself:

$$\mathbf{u}\left(\mathbf{r}_p, t-\Delta t\right)\cdot\nabla\mathbf{u}\left(\mathbf{r}_p, t-\Delta t\right)=0 \tag{32}$$

which performs convective extension, thus assigning the interfacial velocity properly. We should be aware that Eq. (32) does not satisfy the normal zero gradient boundary conditions, and thus all the extensions in air during a given time-step must be carried out using Eq. (15). Once the time-step is converged, the extension of Eq. (32) is performed and loaded, only in air, as the previous time-step velocity.

IV. IMPLEMENTATION

The single-phase level set model was implemented in the code CFDShip-Iowa [31], a parallel unsteady RANS code. The code uses body-fitted structured multi-block grids with ghost cells and chimera interpolations to accommodate complex geometries. The equations are first transformed from the physical (x, y, z, t) domain to the non-orthogonal computational domain (ξ, η, ζ, τ). The resulting equations in water are:

$$\frac{\partial U_i}{\partial \tau}+\frac{1}{J}b_j^k\left(U_j-\frac{\partial x_j}{\partial \tau}\right)\frac{\partial U_i}{\partial \xi^k}=-\frac{1}{J}b_i^k\frac{\partial p}{\partial \xi^k}+\frac{1}{J}\frac{\partial}{\partial \xi^j}\left(\frac{b_i^j b_i^k}{J\,\mathrm{Re}_{eff}}\frac{\partial U_i}{\partial \xi^k}\right)+S_i \tag{33}$$

$$\frac{1}{J}\frac{\partial}{\partial \xi^j}\left(b_i^j U_i\right)=0 \tag{34}$$

$$\frac{\partial \phi_i}{\partial \tau}+\frac{1}{J}b_j^k\left(U_j-\frac{\partial x_j}{\partial \tau}\right)\frac{\partial \phi_i}{\partial \xi^k}=0 \tag{35}$$

The convective terms are discretized using a second-order upwind scheme and the time derivatives are discretized using Euler second-order backward differences. This discretization scheme also applies to the grid velocity terms in Eqs. (33) and (35). The viscous terms in Eq. (33) are discretized using second-order central differences. Similar differences schemes are used to discretize the turbulence equations.

The incompressibility constraint is enforced using the PISO algorithm [32]. Upon discretization, the *i*-velocity component at node *ijk* can be written as:

$$U_i = -\frac{\sum_{nb} a_{nb} U_{i,nb} - S_i}{a_{ijk}} - \frac{b_i^k}{J a_{ijk}} \frac{\partial p}{\partial \xi^k} \tag{36}$$

with a_{ijk} and a_{nb} the pivot and neighbor coefficients of the discretized momentum equations. Enforcing the continuity equation, Eq. (34), results in a Poisson equation for the pressure:

$$\frac{\partial}{\partial \xi^j}\left(\frac{b_i^j b_i^k}{J a_{ijk}} \frac{\partial p}{\partial \xi^k}\right) = \frac{\partial}{\partial \xi^j} \frac{b_i^j}{a_{ijk}} \left(\sum_{nb} a_{nb} U_{i,nb} - S_i\right) \tag{37}$$

Notice that in curvilinear, non-orthogonal grid systems, Eq. (37) leads to a 19-point stencil. In order to avoid pressure-velocity decoupling, the contravariant pressure gradients in Eq. (37) must be evaluated at the faces of the cells on the computational domain. This forces the metric coefficients to be available at half-cell locations, and these are computed directly to prevent the appearance of artificial mass sources if averages are performed [33].

In air, the velocity is extended from the air/water interface using Eq. (15) in discretized form:

$$b_j^k n_j \frac{\partial U_i}{\partial \xi^k} = 0 \tag{38}$$

which enforces the velocity boundary condition at the interface and also provides a velocity field to transport the level set function. In addition, the same extension procedure is performed for the turbulence quantities *k* and ω. The reinitialization of the level set function as a distance, Eq. (24), is discretized similarly to Eq. (38) adding the corresponding source term, and solved iterating a few times.

The convective extension, Eq. (32), is discretized as the convective terms in Eq. (33), dropping all the other terms. This is solved only at the end of each time step and the resulting velocity loaded on the points in air as the previous time-step velocity. We note that performing the convective extension on the turbulence quantities has little effect on the results, but is very important for the velocities.

The resulting algebraic systems for the variables (u, v, w, p, ϕ, k, w) are solved in sequential form and iterated within each time step until convergence. For the pressure Poisson equation, a matrix system is built and solved using the PETSc toolkit [34], while all the other systems are solved using the ADI method.

V. EXAMPLES

Since the aim of this paper is an unsteady method, we concentrate in unsteady example problems. The method has been tested on steady-state three-dimensional problems, including the flow around a surface ship model DTMB 5415 for different Froude numbers [35].

The numerical method described in the previous sections has been applied to three unsteady cases: a two-dimensional linear progressive wave, a viscous wave in a two-dimensional tank and a three-dimensional wave diffraction problem around a surface combatant.

A. Linear Progressive Wave

For analysis purposes, an ideal linear plane progressive wave is attractive because it has an exact analytical solution and therefore allows for direct comparison with the numerical method. A small amplitude wave of the form:

$$\zeta(x,t) = A\sin(kx - \omega t) \tag{39}$$

with k the wavenumber and ω the encounter frequency, was imposed on a two-dimensional domain as initial condition ($t = 0$). Eq. (39) is also the exact solution of the elevation as a function of time for comparison purposes, see for instance [36]. Initial velocities and pressure are also imposed according to the exact solution. The domain extends vertically from $z = -1.5$ to $z = 0.1$ and horizontally from $x = 0$ to $x = x_{\max}$. At $x = 0$ we use Eq. (39) to impose the level set inlet boundary condition as:

$$\phi(0,t) = A\sin(-\omega t) - z \tag{40}$$

Other inlet boundary conditions follow the exact solution. In our case, the wave amplitude is $A = 0.004$, small enough to ensure negligible finite depth effects. The wavelength λ is set to 1, resulting in a wavenumber $k = 2\pi/\lambda = 2\pi$ or a small steepness

$Ak = 0.025$, which guarantees linear wave behavior. The Froude number is $Fr = 0.5$, and the Reynolds is set to $\mathrm{Re} = 10^9$ so that viscous effects are negligible. This results in a phase velocity $V_p = 1 + \sqrt{\lambda / 2\pi} / Fr = 1.798$. An absorption condition (numerical beach) is imposed at the exit to avoid spurious amplitude oscillations. This is implemented damping the level set function of Eq. (6) as:

$$\frac{\partial \phi}{\partial t} + \mathbf{v} \cdot \nabla \phi = -\beta \left(\phi + z \right) \tag{41}$$

where β is a function that is zero everywhere except on the damping region where it grows quadratically from zero at $x = x_d$ to a maximum of 10 at the exit.

Fig. 4 shows a comparison of the wave predicted by the convective extension explained on the previous section with the normal extension for the previous time step velocity. In this case the domain extends to $x_{\max} = 9$ and the numerical beach is activated starting at $x_d = 5$. The grid has 4 blocks, each block 136 grid points in the *x*-direction, covering each wavelength by 36 grid points, by 70 in *z*, conveniently clustered around the wave amplitude. 8 non-dimensional time units were run with constant time step of 0.01 to cover about 14 wave periods.

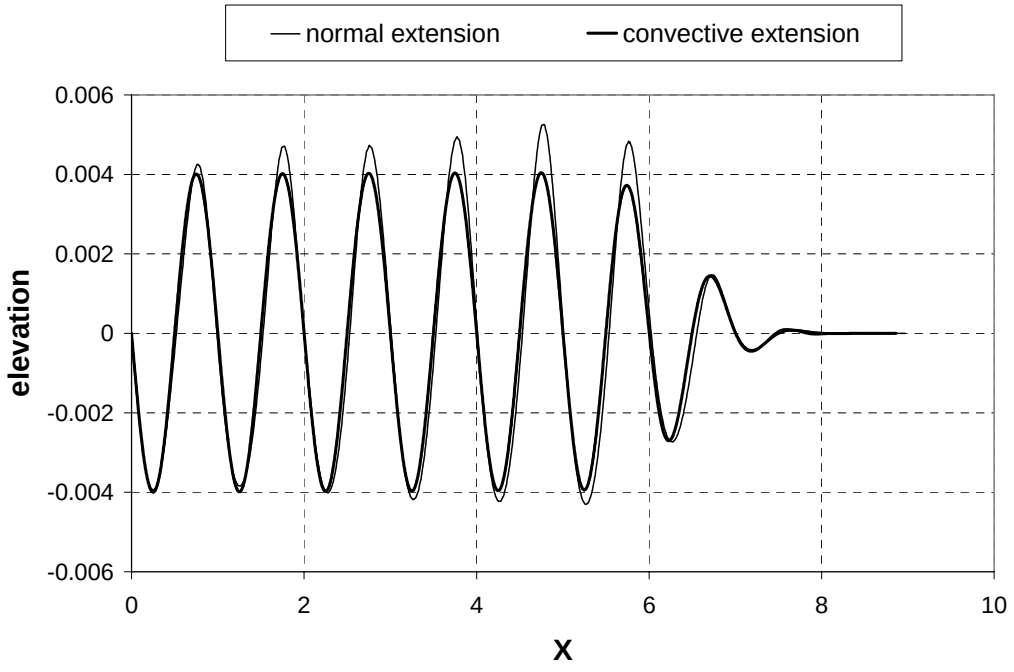

Figure 4: Comparison of normal extension with convective extension for a progressive wave.

It is clear that the normal extension tends to grow the wave as it travels in the positive x direction, while the proposed extension maintains the wave shape appropriately. To make a quantitative comparison with the exact solution we run a similar case but with a longer domain, extending to $x_{\max} = 15$ and damping the free surface starting at $x_d = 10$. This covers 10 wavelengths.

Fig. 5 shows a plot of the numerical and exact solutions, this latter up to $x_d = 10$ since the exact solution does not apply in the damped region. The wave phase is followed correctly, with an error of only $\Delta x = 0.00655$ (0.655 %) at the end of the computation on the crossing by zero at the tenth wavelength. The wave deforms and the amplitude decreases slowly as the wave progresses. At the last wave before the numerical beach is activated, the maximum of the wave has decreased 1 % and the minimum has 2.8 % error. The RMS difference between the numerical and the exact solutions gives an idea of the deformation of the wave, in this case 1.9 %. Following an initial transient, to allow the first wave entering the domain to reach the numerical beach, there is no significant change with time on these errors. This shows that some level of numerical diffusion is affecting the computation, but to acceptable levels for most applications.

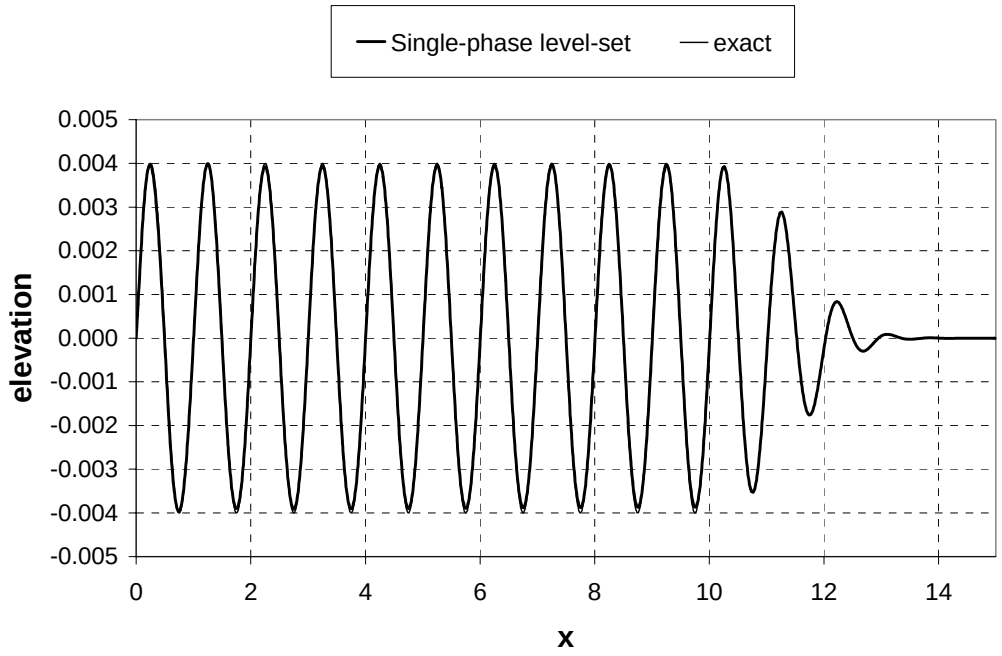

Figure 5: Exact and numerical solution of a progressive linear wave.

B. Sloshing in a Fixed Rectangular Tank

We consider a tank with a length that is twice the depth of the still water level, in which a viscous fluid is allowed to oscillate freely. The grid is comprised of 4 blocks each with 51x46 grid points in the x and z directions respectively and the extents are $x \in (-1, 1)$ and $z \in (-1, 0.1)$ (see Fig. 6). At $t = 0$ the free surface has a sinusoidal profile of small amplitude ζ_0 and wavelength $2d$, which in our case is represented by:

$$\zeta_0(x) = 1 - 0.01\sin\left(\frac{\pi x}{2}\right) \tag{42}$$

The free surface is then released and the wave elevation shows an amplitude decay in time $\zeta(x,t)$. In order to simulate an infinite wave, we impose slip conditions at the lateral walls and at the bottom of the computational domain. Since the velocities are very small at the bottom, the boundary condition there has very little effect.

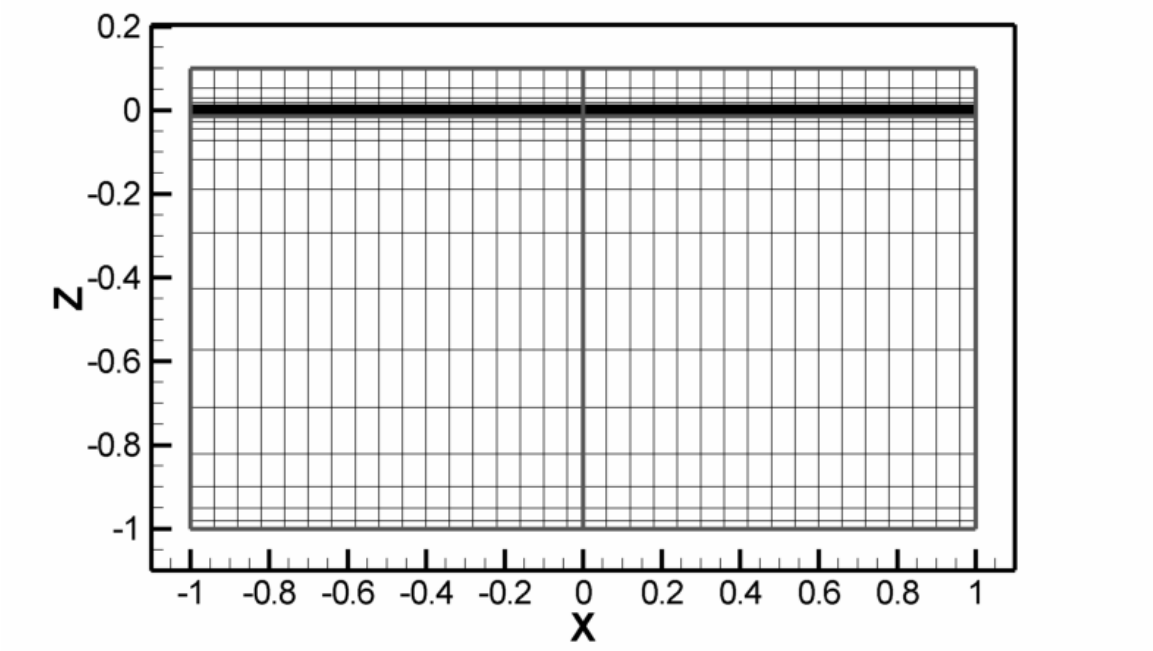

Figure 6: Two-dimensional grid used for the tank case. The bold gray lines indicate the limits of the blocks. One every third grid point is shown for clarity.

This problem was studied analytically by Wu *et al.* [37], who solved the linearized Navier-Stokes equations. The solutions are expressed for different (small) Reynolds numbers ($\mathrm{Re} = d\sqrt{gd}/\nu$) as a function of a dimensionless time expressed as

$\tau = t\sqrt{g/d}$. In our method we obtain the same non-dimensionalization by setting $Fr = 1$ and using the same initial amplitude as in the analytical problem. In addition, Eatock Taylor *et al*. [38] provide numerical solutions to this problem using a Pseudo-Spectral Matrix Element Method, which is deemed to be very accurate, though the authors use linearized free surface conditions.

Fig. 7 shows the free surface evolution for different non-dimensional times for a problem with Re=100. To evaluate quantitatively the performance of our numerical method, we compare against the analytical and the pseudo-spectral method solutions [38] in Fig. 8. We see that the single-phase method does an excellent job in predicting both the amplitude and phase of the analytical solution outperforming the pseudo spectral method predictions.

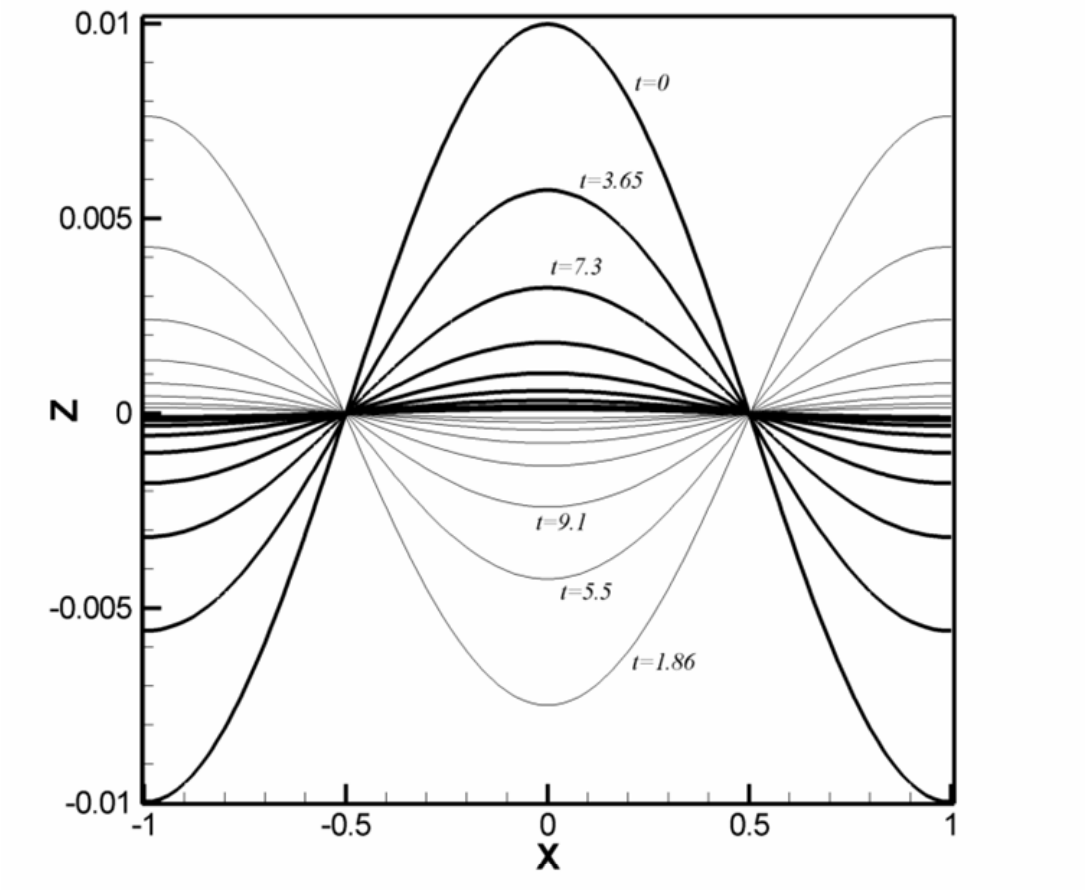

Figure 7: Free surface elevation evolution in a tank for different times at Re=100.

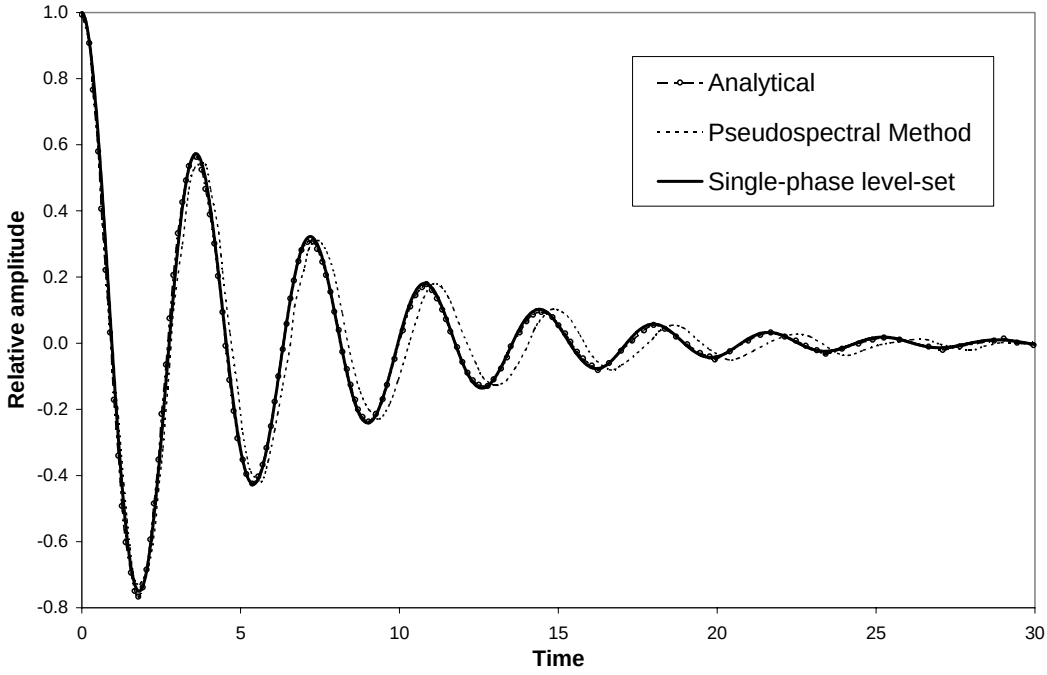

Figure 8: wave amplitude evolution at the center in a two-dimensional tank (Re=100)

The solution for a higher Reynolds number (Re=2000) is shown in Fig. 9, compared against the analytical solution reported in [37]. At this Reynolds number, the single-phase level set method shows slight phase and amplitude differences. We note that the analytical solution neglects the nonlinear terms in the Navier-Stokes equations and uses a linearized free surface boundary condition, which might lead to error at this Reynolds number. This error, however, has not been quantified.

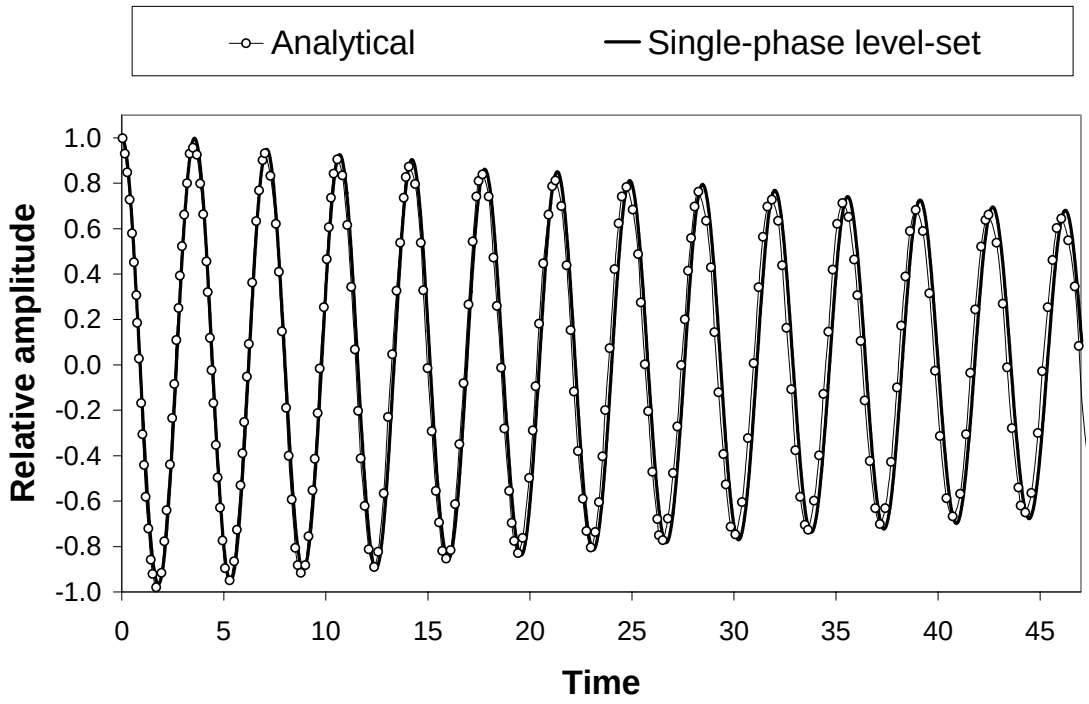

Figure 9: wave amplitude evolution at the center in a two-dimensional tank (Re=2000)

C. Forward Speed Diffraction in a Surface Ship

This problem is attractive as a benchmark because it involves considerable complications with respect to the previous two cases. In this problem a ship is moving with constant speed in the presence of regular head waves, that is, the ship and the waves move in opposite directions. Wilson & Stern [7] and Rhee & Stern [8] have performed numerical simulations of this problem using surface-tracking methods, and Cura Hochbaum and Vogt [20] have used the two-phase level set method.

The test case chosen has been experimentally studied extensively for a DTMB 5512 model and unsteady free surface elevation, resistance, heave force and pitch moments [39,40] and velocities [41] were measured. Though the authors measured forces and moments for a wide range of test conditions, free surface elevations and velocities were measured at medium Froude number ($Fr = 0.28$), long wavelength ($\lambda = 1.5L$) and low wave steepness ($Ak = 0.025$). Thus, these conditions were selected for comparison

with the numerical method. Since the model ship has a length $L = 3.048m$, a Froude number $Fr = 0.28$ corresponds to $\mathrm{Re} = 4.65\ 10^6$.

In order to simulate the boundary layer turbulence using the $k-\omega$ model, a fine near wall discretization is necessary, with spacing around $10^{-6} L$. This makes the design of the computational grid difficult, since an orthogonal grid is convenient in the far field to avoid deformation of the incoming wave. This problem can be avoided using a body-fitted grid for the boundary layer and an orthogonal grid for the far-field using overset grids with Chimera interpolation. In our case we chose to use an 8 block boundary layer body-fitted grid, a 16 block close-field orthogonal grid and a 8 block far-field orthogonal grid, for a total of 32 blocks and approximately 2,000,000 grid points. The interpolation coefficients for the overset grids were generated using Pegasus [41]. The overall grid, shown in Fig. 10, extends from $x = -1$ to $x = 2$, $y \in (0,1)$, taking advantage of the symmetry of the problem about the centerplane $y = 0$, and $z \in (-1, 0.1)$, with the ship located between $x = 0$ to $x = 1$. Ghost cells were used for interblock coupling inside each of the three main grid systems.

The initial conditions are set to those for a progressive wave, similar to the first example presented in this paper but in this case the wave amplitude is $A = 0.006$, in accordance to the experimental conditions. No numerical damping was used at the exit. The boundary conditions are summarized in the table below.

	ϕ	p	k	ω	U	V	W
inlet ($x = -1$)	Eq. (40)	exact solution	$k_{fs} = 10^{-7}$	$\omega_{fs} = 9$	exact solution	$V = 0$	exact solution
exit ($x = 2$)	$\frac{\partial \phi}{\partial n} = 0$	$\frac{\partial p}{\partial n} = 0$	$\frac{\partial k}{\partial n} = 0$	$\frac{\partial \omega}{\partial n} = 0$	$\frac{\partial^2 U}{\partial n^2} = 0$	$\frac{\partial^2 V}{\partial n^2} = 0$	$\frac{\partial^2 W}{\partial n^2} = 0$
far-field ($y = 1$)	$\frac{\partial \phi}{\partial n} = 0$	$\frac{\partial p}{\partial n} = 0$	$\frac{\partial k}{\partial n} = 0$	$\frac{\partial \omega}{\partial n} = 0$	$\frac{\partial U}{\partial n} = 0$	$\frac{\partial V}{\partial n} = 0$	$\frac{\partial W}{\partial n} = 0$
far-field ($z = -1$)	$\frac{\partial \phi}{\partial n} = 1$	$\frac{\partial p}{\partial n} = 0$	$\frac{\partial k}{\partial n} = 0$	$\frac{\partial \omega}{\partial n} = 0$	$U = 1$	$V = 0$	$W = 0$
symmetry ($y = 0$)	$\frac{\partial \phi}{\partial n} = 0$	$\frac{\partial p}{\partial n} = 0$	$\frac{\partial k}{\partial n} = 0$	$\frac{\partial \omega}{\partial n} = 0$	$\frac{\partial U}{\partial n} = 0$	$V = 0$	$\frac{\partial W}{\partial n} = 0$
far-field ($z = 0.1$)	$\frac{\partial \phi}{\partial n} = -1$	not needed	$\frac{\partial k}{\partial n} = 0$	$\frac{\partial \omega}{\partial n} = 0$	$\frac{\partial U}{\partial n} = 0$	$\frac{\partial V}{\partial n} = 0$	$\frac{\partial W}{\partial n} = 0$
no slip (ship wall)	$\frac{\partial \phi}{\partial n} = 0$	Eq. (37)	$k = 0$	$\omega = \frac{60}{\mathrm{Re}\,\beta y^{+2}}$	$U = 0$	$V = 0$	$W = 0$

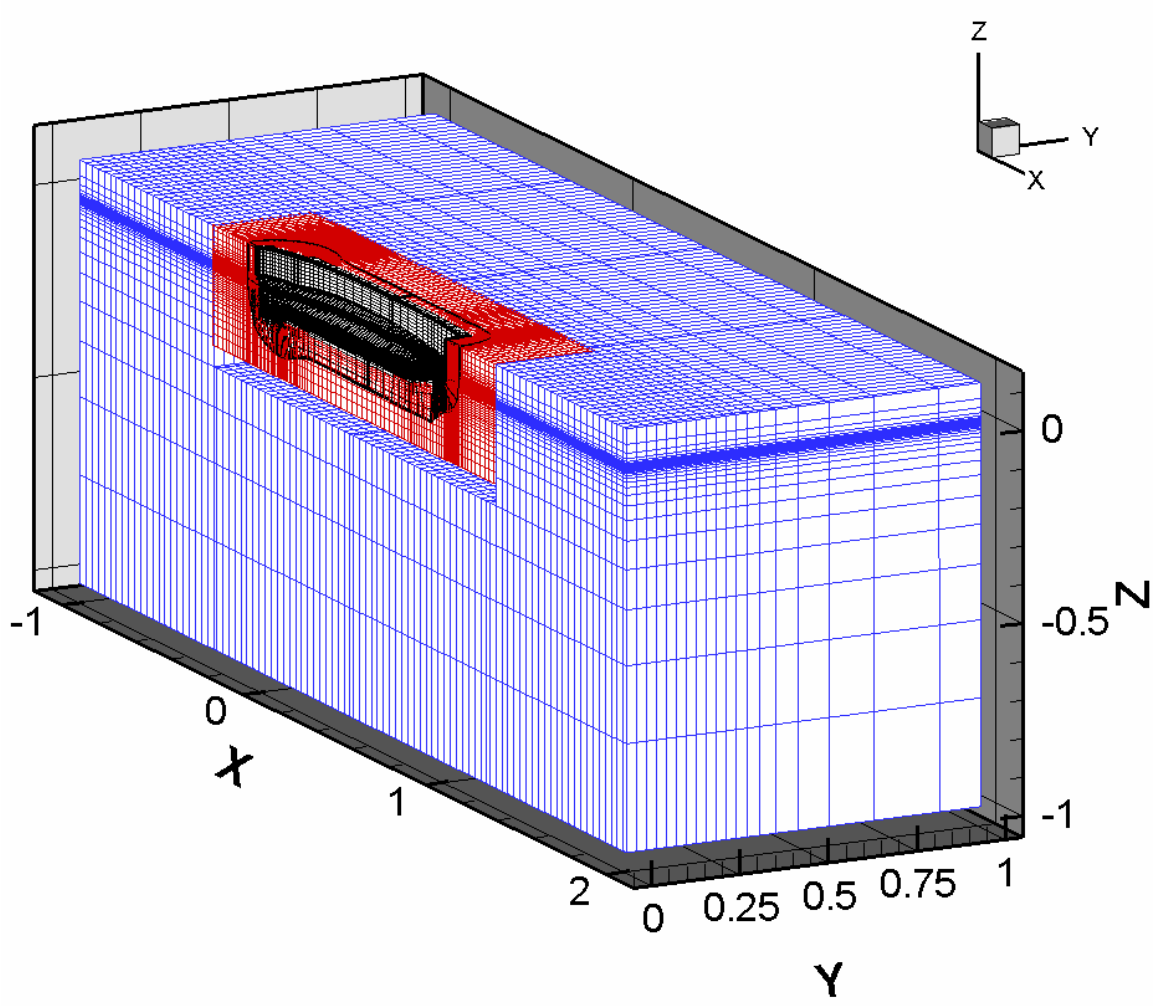

Figure 10: multi-block overset grid for the wave diffraction problem. One every other point shown for clarity.

The computation was started at $t = 0$, with sudden imposition of the boundary conditions, which causes an acceleration transient. The time step was chosen to be 0.00683, so that each wave period was discretized in 80 time steps. The initial transient is due to the time needed for the ship boundary layer to grow and for the Kelvin waves to develop, but an essentially periodic solution was achieved after about 3 non-dimensional time units, equivalent in our non dimensionalization to an advance of three ship lengths L. After the periodic solution was achieved, 5 more periods were run.

We will concentrate in this paper on comparisons with free surface elevations and wake velocities. Since the behavior at the experimental conditions is linear, the free

surface elevations were reconstructed and reported in terms of the zero and first Fourier harmonic amplitudes and first Fourier phase [38,39]. We chose to compare against quarter periods at $t/T = 0, 1/4, 1/2$ and 3/4. The phase is set in such a way that at the beginning of the period T the crest of the wave is coincident with the bow of the ship, $x = 0$. Since the wavelength is $\lambda = 1.5L$, the far-field crest will be located at $x = 0.375$, 0.75 and 1.125 for $t/T = 1/4, 1/2$ and 3/4, respectively.

The free surface elevation predicted by the single-phase level-set method is compared against the experimental data and surface-tracking simulations performed by [7] for the periodic state at t/T = 0 in Fig. 11. The single-phase level set results show an excellent agreement with the experimental data capturing appropriately the Kelvin waves and the near-hull features of the free surface. Compared to the surface-tracking results, the single-phase level set shows much better agreement with the experimental data. It must be stressed that the surface-tracking computations were made on a coarser grid (750,000 grid points) and that could partially account for the poorer results. Notice, however, that the implementation of overset grids with surface tracking methods would be complicated and expensive, since the interpolation coefficients would need to be recomputed every time the grid moves, which means several times per time step and at every time step in an unsteady problem. Thus, in surface capturing methods much better quality grids can be used.

Fig. 12 shows the elevations from the experimental data and the level set computations for the other three quarter periods: $t/T = 1/4, 1/2$ and 1/4. Again, excellent agreement with the experimental data is evident, though the elevation gradients appear to be slightly smoothed, resulting in some underprediction of the crests and troughs on the Kelvin wave. Contrarily, the wave elevation on the stern is slightly overpredicted.

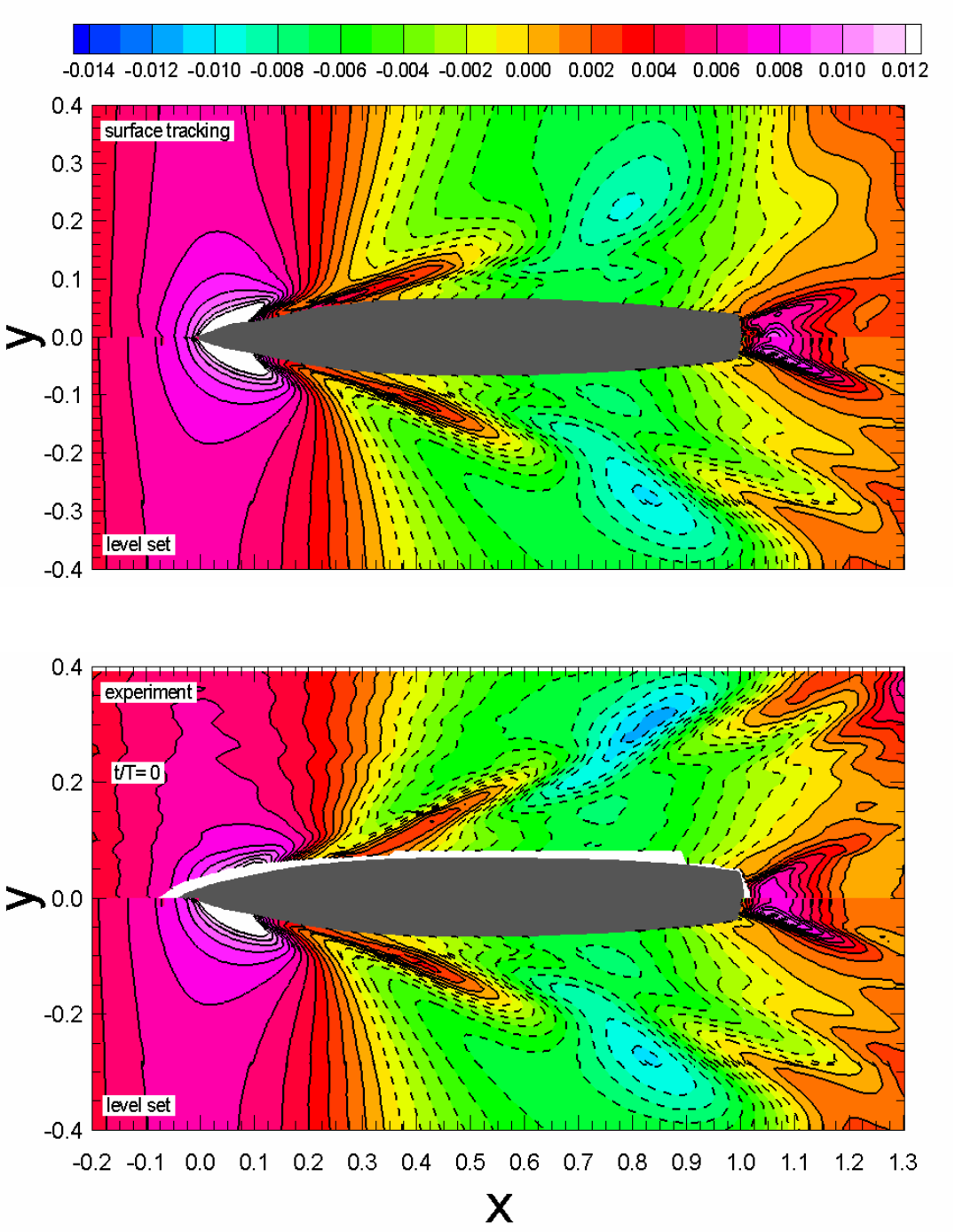

Figure 11: Free surface contours for $t/T = 0$. The single-phase level set results are compared against surface tracking computations and experimental data.

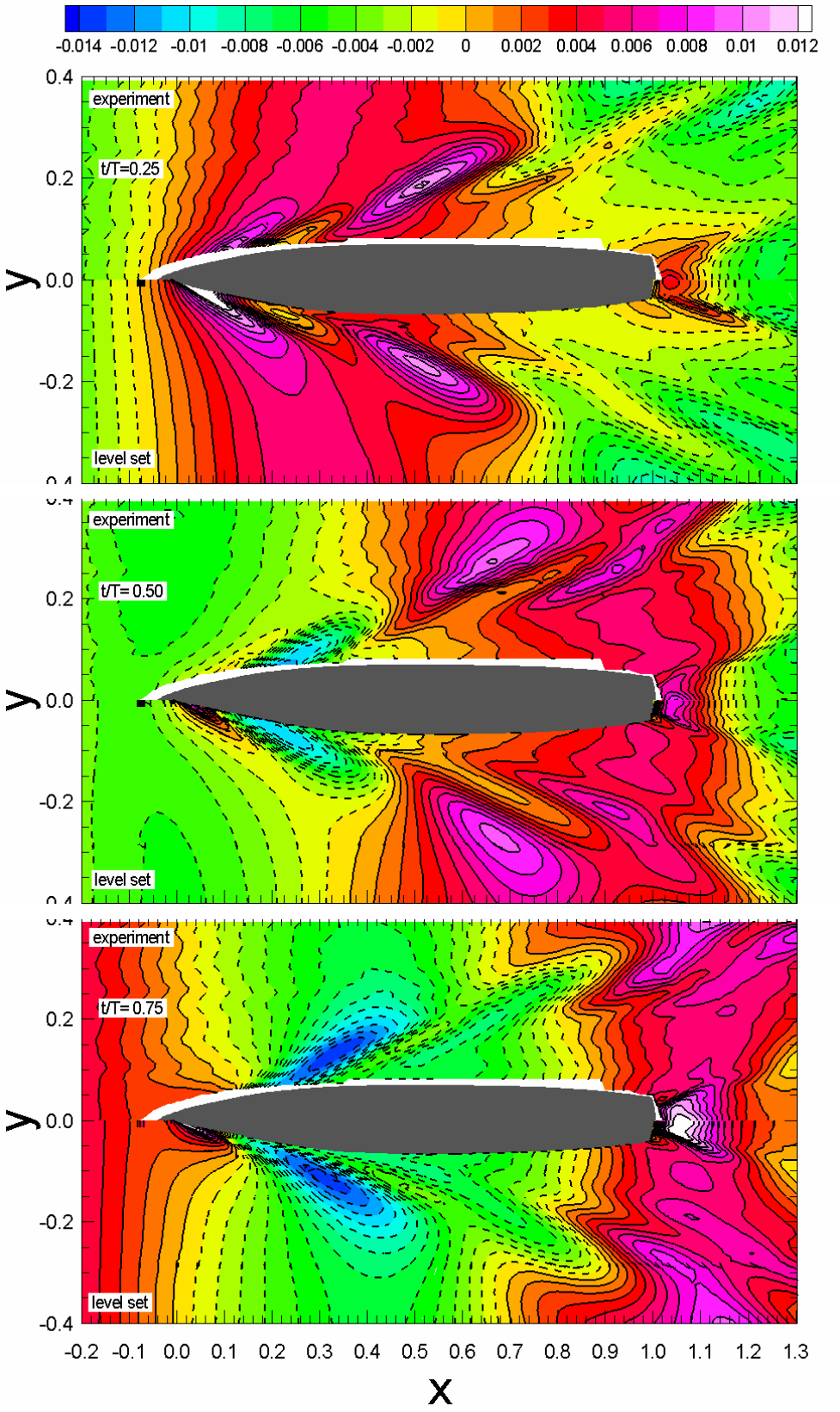

Figure 12: Free surface contours for t/T = 1/4, 1/2 and 3/4. Port: single-phase level set, starboard: experimental data.

PIV data of the velocity field at the nominal wake plane ($x/L = 0.935$) has been obtained by Longo *et al.* [40] using a phase-averaging technique. In Figs. 13, 14, 15, and 16 we compare the single-phase level set method results against surface tracking predictions and experimental data. Though the results are reasonably good and better than those obtained with the surface-tracking method, the boundary layer thickness is apparently to some extent overpredicted and the *V* and *W* velocities show that the vortex detached from the sonar dome has lost more strength on the computations than on the experiments. This trend could be due to the two-equation turbulence model used in the computations that cannot capture anisotropy on the Reynolds stresses.

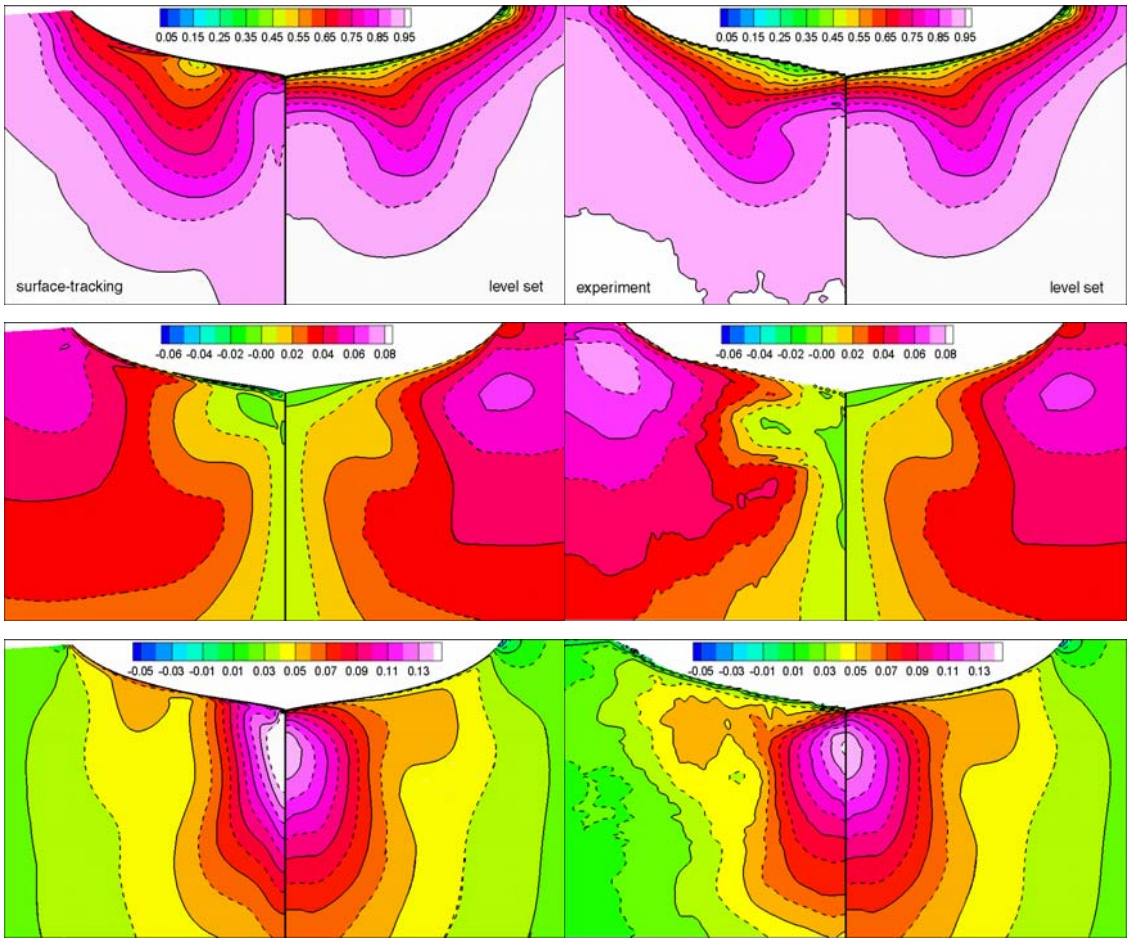

Figure 13: Velocity contours at the nominal wake plane for t/T = 0. U, V and W are shown on the upper, center and lower figures, respectively. Left side: surface tracking (port) vs. single-phase level-set (starboard). Right side: experimental data (port) vs. single-phase level-set (starboard).

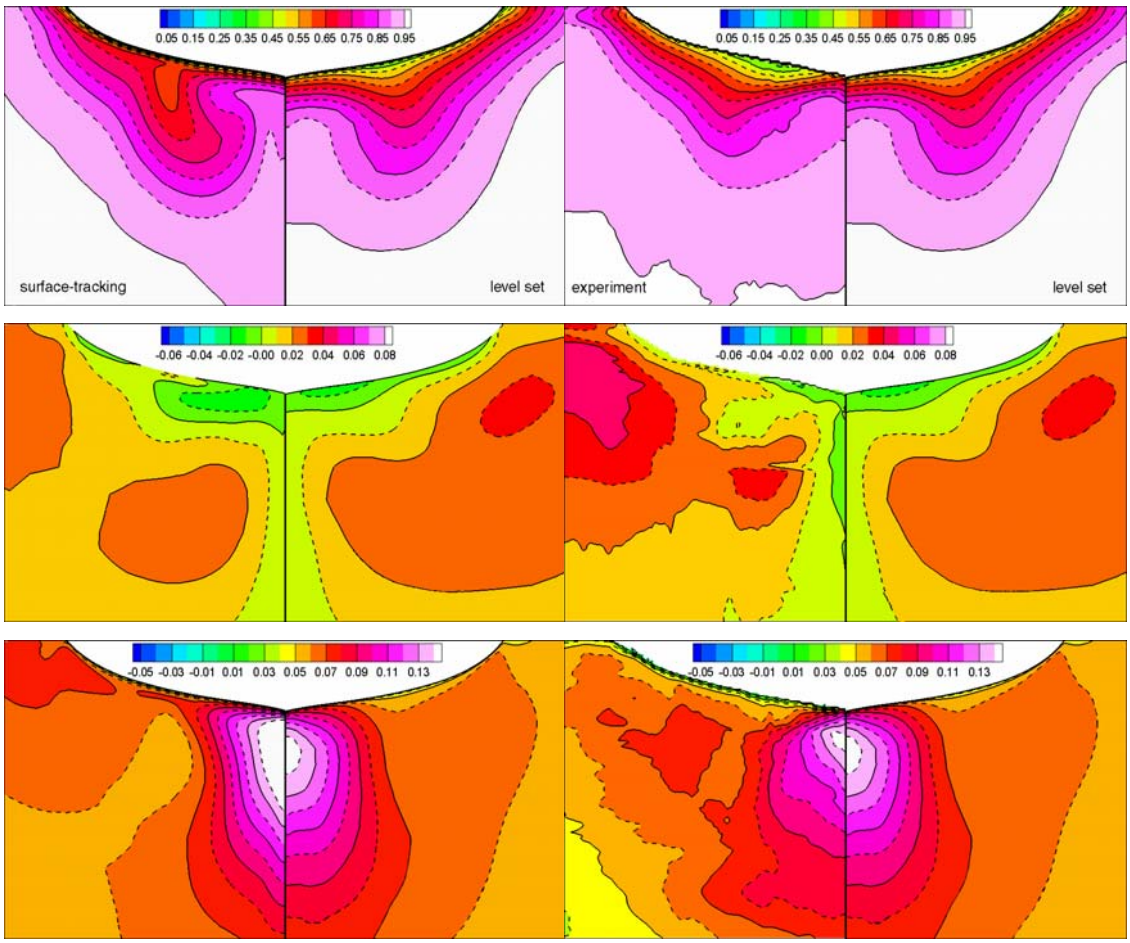

Figure 14: Velocity contours at the nominal wake plane for t/T = 1/4. U, V and W are shown on the upper, center and lower figures, respectively. Left side: surface tracking (port) vs. single-phase level-set (starboard). Right side: experimental data (port) vs. single-phase level-set (starboard).

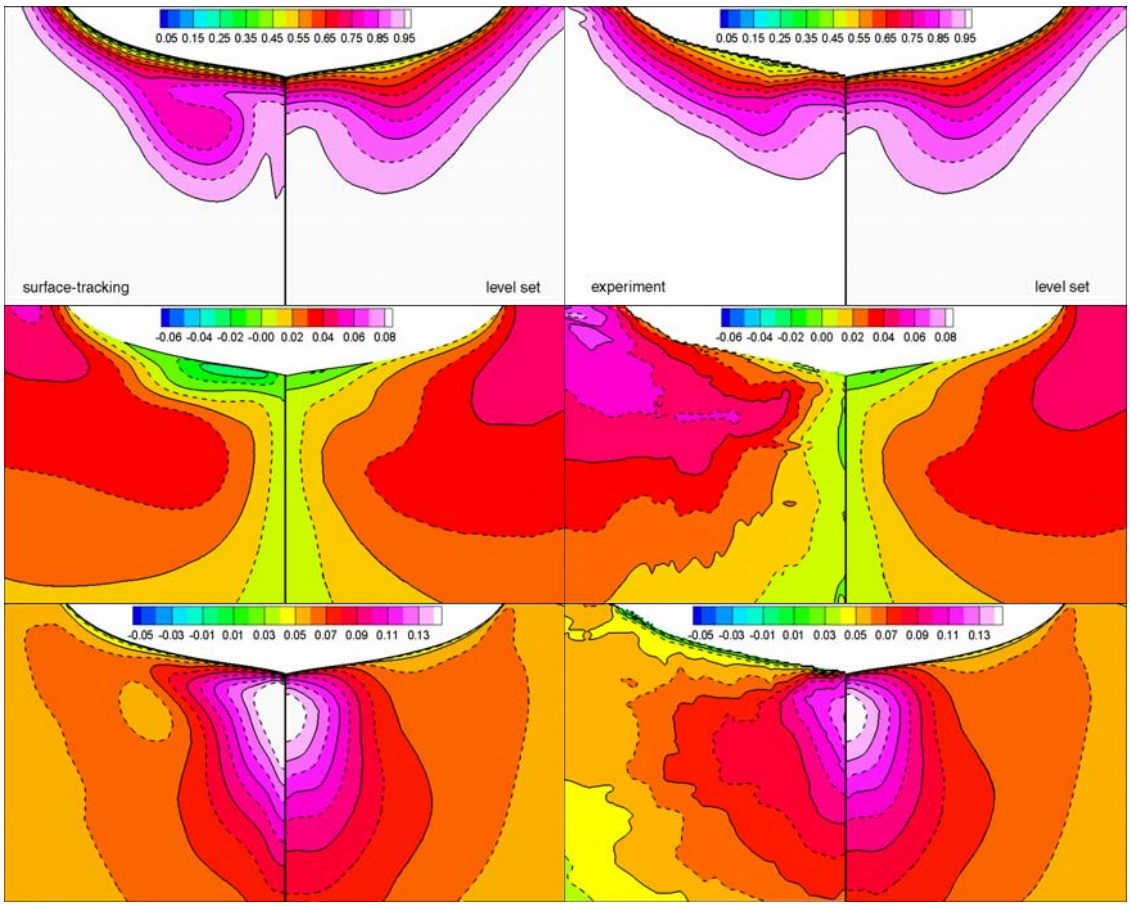

Figure 15: Velocity contours at the nominal wake plane for t/T = 1/2. U, V and W are shown on the upper, center and lower figures, respectively. Left side: surface tracking (port) vs. single-phase level-set (starboard). Right side: experimental data (port) vs. single-phase level-set (starboard).

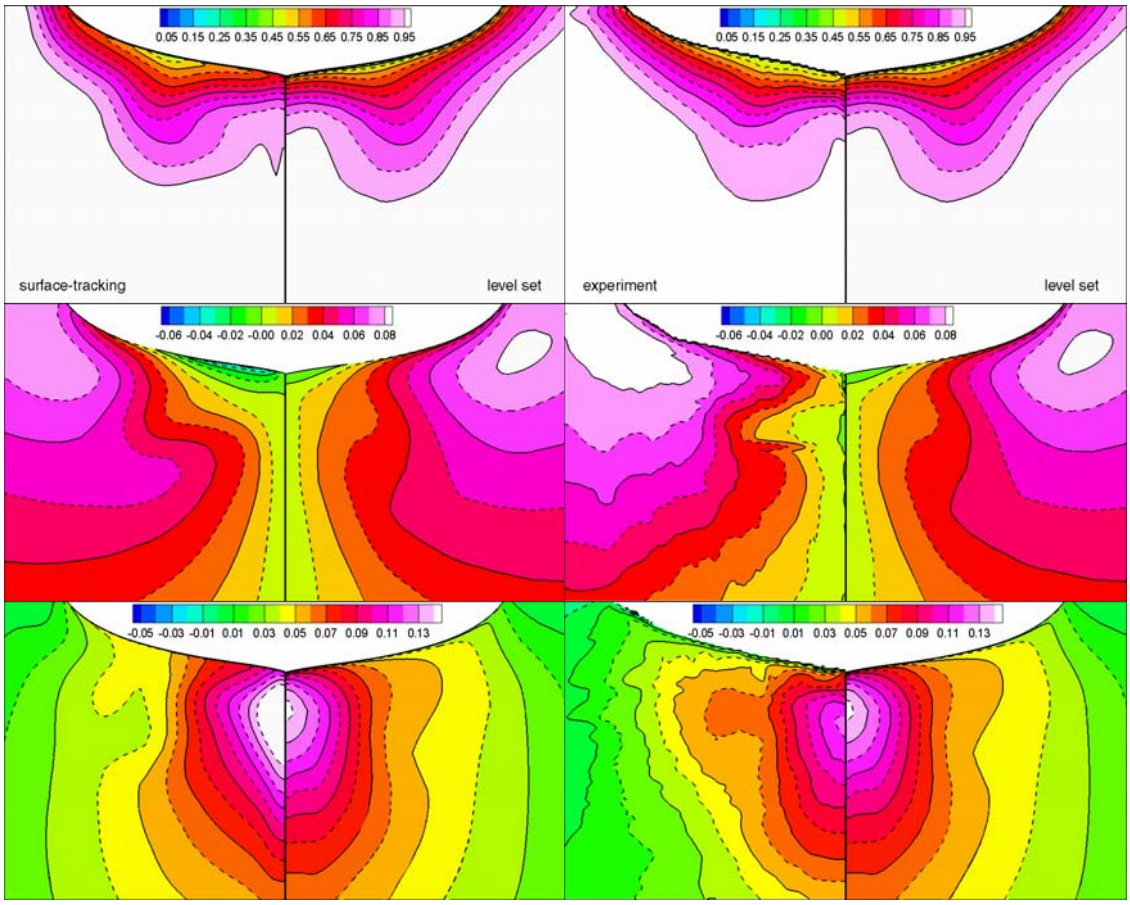

Figure 16: Velocity contours at the nominal wake plane for t/T = 3/4. U, V and W are shown on the upper, center and lower figures, respectively. Left side: surface tracking (port) vs. single-phase level-set (starboard). Right side: experimental data (port) vs. single-phase level-set (starboard).

VI. CONCLUSIONS

An unsteady single-phase level-set method has been presented. The method relies on the level set function to detect the interface, and on velocity extensions and pressure interpolations to enforce the jump conditions at the interface. The computation of the time derivatives to properly evaluate the total derivatives is discussed in detail. The method is tested against three unsteady cases: an inviscid linear progressive wave, the viscous sloshing in a two-dimensional tank and the flow around a surface ship under regular head waves. In the three cases the method has performed very well compared to either analytical or experimental results.

The presented method has several potential advantages against the standard two-phase level set method, and of course some disadvantages. One of the advantages is that the computation takes place only in water, with potential important savings computing simpler equations in air. In addition, the computation is performed in a fluid with constant properties, avoiding the problem related with large density ratios in two-phase level set methods. Since the jump conditions are imposed explicitly, no transition zone appears. As for the shortcomings, the method does not solve the fluid equations in air and therefore no problems in which air entrapment occurs can be solved. In addition, the stresses on the liquid caused by the air must be negligible or specified somehow.

In principle any problem that is not restricted by the limitations previously stated can be tackled with the unsteady single-phase level-set method. In particular, this includes surface-piercing bodies with large-amplitude waves or motions, very steep waves, etc. Linear and nonlinear problems can be solved since we retain the nonlinear terms on the equations and on the jump conditions.

We have demonstrated the capability of the single-phase level set method to solve complex transient free surface three-dimensional problems. The future work includes an extensive analysis of the forward speed wave diffraction problem, including quantitative verification and validation and study of linear and nonlinear behavior of forces and moments for larger Froude numbers and shorter wavelength. Extensions of interest that require 6DOF capability are the prediction of large-amplitude motions, including pitching and heaving and free and forced rolling, and maneuverings.

ACKNOWLEDGEMENTS

This research was sponsored by the Office of Naval Research under Grant N00014-01-1-0073. Dr. Patrick Purtell was the program manager.

REFERENCES

1. J. A. Sethian and P. Smereka, Level set Methods for Fluid Interfaces, *Annu. Rev. Fluid Mech.* 2003; **35**: 341-372.
2. S. J. Osher and J. A. Sethian, Front Propagating with Curvature Dependent Speed: Algorithms Based on Hamilton-Jacobi Formulations, *J. Comput. Phys*. 1988; **79**: 12-49.
3. S. J. Osher and R. P. Fedkiw, Level set Methods: an Overview and Some Recent Results, *J. Comput. Phys*. 2001; **169**: 463-502.
4. M. S. Kim and W. I. Lee, A New VOF-Based Numerical Scheme for the Simulation of Fluid Flow with Free Surface. Part I: New Free Surface-Tracking Algorithm and its Verification, *Int. J. Num. Meth. Fluids* 2003; **42**: 765-790.
5. R. Wilson, F. Stern, H. Coleman and E. Paterson, Comprehensive Approach to Verification and Validation of CFD Simulations-Part 2: Application for RANS Simulation of a Cargo/Container Ship, *ASME J. Fluids Eng*. 2001; **123**: 803-810.
6. C. O. E. Burg, K. Sreenivas, and D. G. Hyams, Unstructured Nonlinear Free Surface Simulations for the Fully-Appended DTMB Model 5415 Series Hull Including Rotating Propulsors, *24th ONR Symp. on Naval Hydrodynamics*, Fukuoka, Japan 2002.
7. R. V. Wilson and F. Stern, Unsteady CFD Method for Naval Combatants in Waves, *22nd ONR Symp. on Naval Hydrodynamics*, Washington DC, USA 1998.
8. S. H. Rhee and F. Stern, Unsteady RANS Method for Surface Ship Boundary Layer and Wake and Wave Field, *Int. J. Num. Meth. Fluids* 2001; **37**: 445-478.
9. R. V. Wilson, P. M. Carrica and F. Stern, Unsteady RANS Simulation of a Surface Combatant with Roll Motion, submitted to *Comp. Fluids* (2004)

10. G. Weymouth, R. V. Wilson and Stern, RANS CFD Predictions of Pitch and Heave Ship Motions in Head Seas, accepted for publication in *J. Ship Res*. (2004)

11. L. Larsson, F. Stern and V. Bertram, Benchmarking of Computational Fluid Dynamics for Ship Flows: The Gothenburg 2000 Workshop, *J. Ship Res.* 2003; **47**: 63-81.

12. C. W. Hirt and B. D. Nichols, Volume of Fluid (VOF) Method for Dynamics of Free Boundaries, *J. Comput. Phys*. 1981; **39**: 201-221.

13. S. O. Unverdi and G. Tryggvason, A Front Tracking Method for Viscous, Incompressible, Multi-Fluid Flows, *J. Comput. Phys*. 1992; **100**: 25-37.

14. R. Azcueta, S. Muzaferija and M. Peric, Computation of Breaking Bow Waves For a Very Full Hull Ship, *7th Int. Conf. Numerical Ship Hydrodynamics*, Nantes, France 1999.

15. Y. Sato, H. Miyata and T. Sato, CFD Simulation of 3-Dimensional Motion of a Ship in Waves: Application to an Advancing Ship in Regular Heading Waves, *J. Marine Sci. Tech.* 1999; **4**: 108-116.

16. D. S. Nichols, Development of a Free Surface Method Utilizing an Incompressible Multi-Phase Algorithm to Study the Flow About Surface Ships and Underwater Vehicles, PhD Thesis, Dept. of Aerospace Eng., Mississippi State University, 2002.

17. M. Vogt and L. Larsson, Level set Methods for Predicting Viscous Free Surface Flows, *7th Int. Conf. Numerical Ship Hydrodynamics*, Nantes, France 1999.

18. A. Cura Hochbaum and C. Shumann, Free Surface Viscous Flow Around Ship Models, *7th Int. Conf. Numerical Ship Hydrodynamics*, Nantes, France 1999.

19. A. Cura Hochbaum and M. Vogt, Flow and Resistance Prediction for a Container Ship, *Proc. Gothenburg 2000, A Workshop on CFD in Ship Hydrodynamics*, Gothenburg, Sweden 2000.

20. A. Cura Hochbaum and M. Vogt, Towards the Simulation of Seakeeping and Maneuvering Based on the Computation of the Free Surface Viscous Ship Flow, *24th ONR Symp. on Naval Hydrodynamics*, Fukuoka, Japan 2002.

21. A. Di Mascio, R. Broglia & R. Muscari, A Single-Phase Level set Method for Solving Viscous Free Surface Flows, submitted to *Int. J. Num. Meth. Fluids* (2004)

22. M. Sussman, P. Smereka and S. J. Osher, A Level set Approach to Computing Solutions to Incompressible Two-Phase Flow, *J. Comput. Phys*. 1994; **114**: 146-159 (1994).

23. D. A. Drew and S. L. Passman, *Theory of Multicomponent Fluids*, Springer-Verlag, New York 1998; p. 89.

24. C. Marangoni, Ueber die Ausbreitung der Tropfen einer Flüssigkeit auf der Oberfläche einer Anderen, *Ann. Phys. Chem*. 1871; **143**: 337-354.

25. J. U. Brackbill, D. B. Kothe and C. A. Zemach, A Continuum Method for Modeling Surface Tension, *J. Comput. Phys.* 1992; **100**: 335-354.

26. R. P. Fedkiw, T. Aslam, B. Merriman and S. J. Osher, A non-Oscillatory Eulerian Approach to Interfaces in Multimaterial Flows (the Ghost Fluid Method), *J. Comput. Phys.* 1999; **154**: 393-427.

27. M. Kang, R. P. Fedkiw and L. D. Liu, A Boundary Condition Capturing Method for Multiphase Incompressible Flow, *J. Sci. Comp*. 2000; **15**: 323-360.

28. F. R. Menter, Two-Equation Eddy Viscosity Turbulence Models for Engineering Applications, *AIAA J.* 1994; **32**: 1598-1605.

29. J. A. Sethian, *Level set Methods: Evolving Interfaces in Geometry, Fluid Mechanics, Computer Vision and Material Science*, Cambridge University Press, Cambridge, 1996.

30. D. Adalsteinsson and J. A. Sethian, The Fast Construction of Extension Velocities in Level Set Methods, *J. Comput. Phys.* 1999; **148**: 2-22.

31. E. G. Paterson, R. V. Wilson and F. Stern, General-Purpose Parallel Unsteady RANS Ship Hydrodynamics Code: CFDShip-Iowa, *IIHR report 432*, Iowa Institute of Hydraulic Research, The University of Iowa (2003).

32. R. I. Issa, Solution of the Implicitly Discretized Fluid Flow Equations by Operator Splitting, *J. Comput. Phys*. 1985; **62**: 40-65.

33. J. F. Thompson, Z. U. A. Warsi and C. W. Mastin, *Numerical Grid Generation*, North Holland, Amsterdam, 1992.

34. S. Balay, K. Buschelman, W. Gropp, D. Kaushik, M. Knepley, L. Curfman, B. Smith and H. Zhang, *PETSc User Manual*, ANL-95/11-Revision 2.1.5, Argonne National Laboratory (2002).

35. R. Wilson, P. M. Carrica and F. Stern, Steady and Unsteady Single-Phase Level-Set Method for Large Amplitude Ship Motions and Maneuvering, to be presented at *25th ONR Symp. on Naval Hydrodynamics*, New Foundland, Canada 2004.

36. J. N. Newman, *Marine Hydrodynamics*, MIT Press, Cambridge 1977; p. 240.

37. G. X. Wu, R. Eatock Taylor and D. M. Greaves, The Effect of Viscosity on the Transient Free Surface Waves in a Two-Dimensional Tank, *J. Eng. Math* 2001; **40**: 77-90.

38. R. Eatock Taylor, A. G. L. Borthwick, M. J. Chern and G. Zhu, Modelling Unsteady Viscous Free Surface Flows, *Int. Maritime Res. & Tech. Conf*., Crete, Greece, October 2001.

39. L. Gui, J. Longo, B. Metcalf, J. Shao and F. Stern. Forces, Moment and Wave Pattern for Surface Combatant in Regular Head Waves. Part I: Measurement systems and uncertainty analysis, *Exp. Fluids* 2001; **31**: 674-680.

40. L. Gui, J. Longo, B. Metcalf, J. Shao and F. Stern. Forces, Moment and Wave Pattern for Surface Combatant in Regular Head Waves. Part I: Measurement results and discussion, *Exp. Fluids* 2002; **32**: 27-36.

41. J. Longo, J. Shao, M. Irvine and F. Stern. Phase-Averaged PIV for Surface Combatant in Regular Head Waves, *24th ONR Symp. on Naval Hydrodynamics*, Fukuoka, Japan 2002.

42. N. E. Suhs, S. E. Rogers and W. E. Dietz, Pegasus 5: An Automated Pre-Processor for Overset-Grid CFD, *AIAA paper 2002-3186, 32nd AIAA Fluid Dynamics Conference*, St. Louis 2002.